Linear Regression with Multiple Variables

전 재 욱

Embedded System 연구실
성균관대학교

Outline

- Multiple features
- Gradient descent for multiple variables
- Feature scaling in gradient descent
- Learning rate in gradient descent
- Features and polynomial regression
- Normal equation

Outline

- Multiple features
- Gradient descent for multiple variables
- Feature scaling in gradient descent
- Learning rate in gradient descent
- Features and polynomial regression
- Normal equation

Multiple Features (Variables)

- $h_\theta(x) = \theta_0 + \theta_1 x$

Size in feet^2 (x)	Price ($) in 1000's (y)
2104	460
1416	232
1534	315
852	178
…	…

Multiple Features (Variables)

Size ($feet^2$)	Number of Bedrooms	Number of Floors	Age of home (years)	Price ($1000)
2104	5	1	45	460
1416	3	2	40	232
1534	3	2	30	315
852	2	1	36	178
…	…	…	…	…

Multiple Features (Variables)

Size (feet^2) x_1	Number of Bedrooms x_2	Number of Floors x_3	Age of home (years) x_4	Price ($1000) y
2104	5	1	45	460
1416	3	2	40	232
1534	3	2	30	315
852	2	1	36	178
…	…	…	…	…

- Notation:
  - n : number of features (n =4 in the above)
  - $x^{(i)}$: input (features) of i^{th} training example
  - $x_j^{(i)}$: value of feature j in i^{th} training example

$$x^{(2)} = \begin{bmatrix} 1416 \\ 3 \\ 2 \\ 40 \end{bmatrix}$$

$$x_3^{(3)} = 2$$

Multiple Features (Variables)

- Hypothesis (one variable)
  - $h_\theta(x) = \theta_0 + \theta_1 x$
- Hypothesis (Multiple variables)
  - $h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_3 + \theta_4 x_4$
    - e.g. $h_\theta(x) = 90 + 0.2x_1 + 0.03x_2 + 2x_3 - 3x_4$

Multiple Features (Variables)

- Multivariate linear regression
  - $h_\theta(x) = \theta_0 x_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_3 + \theta_4 x_4$
  - For convenience of notation, define $x_0 = 1$. ($i.e.$ $x_0^{(i)} = 1$).
  - $x = \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \in R^{n+1}$, $\theta = \begin{bmatrix} \theta_0 \\ \theta_1 \\ \theta_2 \\ \vdots \\ \theta_n \end{bmatrix} \in R^{n+1}$

- $$\begin{aligned} h_\theta(x) &= \theta_0 x_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_3 + \theta_4 x_4 \\ &= \theta^T x \\ &= x^T \theta \end{aligned}$$

Outline

- Multiple features
- Gradient descent for multiple variables
- Feature scaling in gradient descent
- Learning rate in gradient descent
- Features and polynomial regression
- Normal equation

Gradient Descent for Multiple Variables

- Hypothesis
  - $h_\theta(x) = \theta^T x = \theta_0 x_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_3 + \theta_4 x_4$
- Parameters
  - $\theta = [\theta_0, \theta_1, \cdots, \theta_n]^T \in R^{n+1}$
- Cost function
  - $J(\theta) = J(\theta_0, \theta_1, \cdots, \theta_n) = \frac{1}{2m}\sum_{i=1}^{m}\left(h_\theta\left(x^{(i)}\right) - y^{(i)}\right)^2$
- Gradient descent
  - Repeat {

$$\theta_j \leftarrow \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta)$$

} (simultaneously update for every $j = 0,1,2, \dots, n$)

Gradient Descent

- One variable ($n = 1$)

- Repeat {

$$\theta_0 \leftarrow \theta_0 - \alpha \frac{1}{m} \sum_{i=1}^{m} \left(h_\theta\left(x^{(i)}\right) - y^{(i)} \right)$$

$$\theta_1 \leftarrow \theta_1 - \alpha \frac{1}{m} \sum_{i=1}^{m} \left(h_\theta\left(x^{(i)}\right) - y^{(i)} \right) x^{(i)}$$

}

(Update for θ_0 and θ_1 simultaneously)

- Mutiple ($n \geq 1$)

- Repeat {

$$\theta_j \leftarrow \theta_j - \alpha \frac{1}{m} \sum_{i=1}^{m} \left(h_\theta\left(x^{(i)}\right) - y^{(i)} \right) x_j^{(i)}$$

}

(Update θ_j for every $j = 0,1,2,\ldots,n$ simultaneously)

$\boxed{x_0^{(i)} = 1}$

$$\theta_0 \leftarrow \theta_0 - \alpha \frac{1}{m} \sum_{i=1}^{m} \left(h_\theta\left(x^{(i)}\right) - y^{(i)} \right) x_0^{(i)}$$

$$\theta_1 \leftarrow \theta_1 - \alpha \frac{1}{m} \sum_{i=1}^{m} \left(h_\theta\left(x^{(i)}\right) - y^{(i)} \right) x_1^{(i)}$$

$$\theta_2 \leftarrow \theta_2 - \alpha \frac{1}{m} \sum_{i=1}^{m} \left(h_\theta\left(x^{(i)}\right) - y^{(i)} \right) x_2^{(i)}$$

…

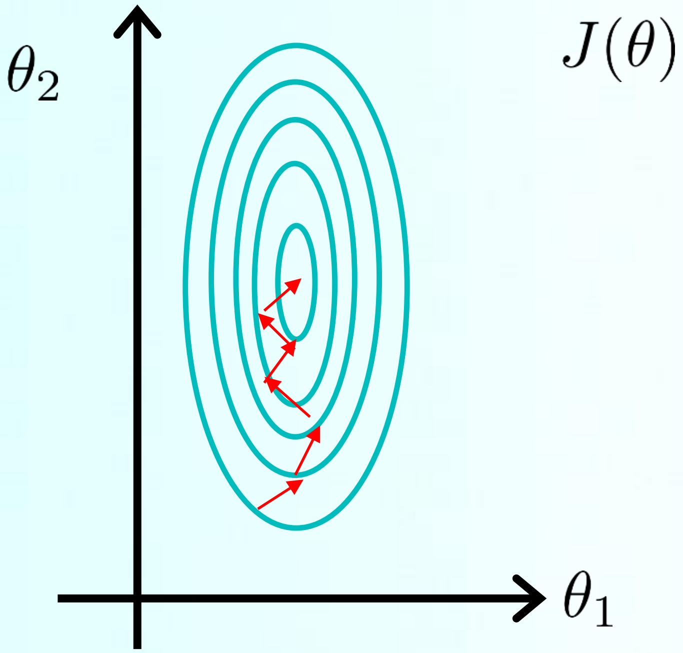
Outline

- Multiple features
- Gradient descent for multiple variables
- Feature scaling in gradient descent
- Learning rate in gradient descent
- Features and polynomial regression
- Normal equation

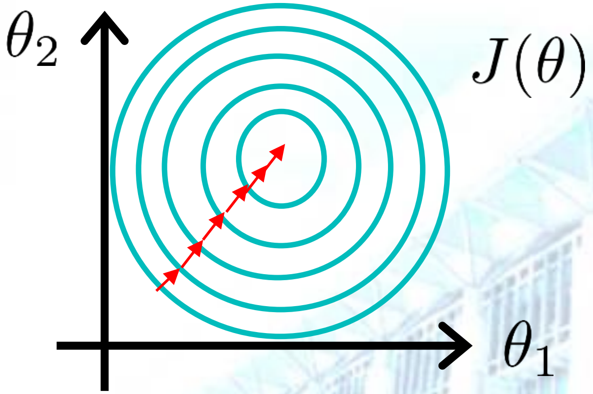
Feature Scaling

- Idea: Make sure features are on a similar scale

- x_1 = size (0 – 2000 feet^2)
- x_2 = number of bedrooms (1 – 5)



- $x_1 = \frac{\text{size (feet}^2\text{)}}{2000}$
- $x_2 = \frac{\text{number of bedrooms}}{5}$
- $0 \leq x_1, x_2 \leq 1$
- Making gradient descent converge much faster



Feature Scaling

Feature Scaling

- Get every feature into approximately $-1 \le x_1, x_2 \le 1$ range.

- Given $x_0 = 1$,
  - $0 \le x_1 \le 3$ ➔ OK
  - $-2 \le x_2 \le 0.5$ ➔ OK
  - $-100 \le x_2 \le 100$ ➔ change
  - $-0.0001 \le x_2 \le 0.0001$ ➔ change

Feature Scaling

- Mean Normalization
  - Replace x_i with $x_i - \mu_i$ to make features have approximately zero mean
    - (Do not apply to $x_0 = 1$).
  - For example,
    - $x_1 = \frac{size - 100}{2000}$, $x_2 = \frac{\#\, of\ bedrooms - 2}{5}$
    - $-0.5 \leq x_1, x_2 \leq 0.5$

- $x_1 \leftarrow \frac{x_1 - \mu_1}{S_1}$
  - μ_1: average value of x_1
  - S_1
    - Either Range of x_1 (max-min) or Standard deviation of x_1

- $x_2 \leftarrow \frac{x_2 - \mu_2}{S_2}$
  - μ_2: average value of x_2
  - S_2
    - Either Range of x_2 (max-min) or Standard deviation of x_2

Outline

- Multiple features
- Gradient descent for multiple variables
- Feature scaling in gradient descent
- Learning rate in gradient descent
- Features and polynomial regression
- Normal equation

Gradient Descent

- $\theta_j \leftarrow \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta)$

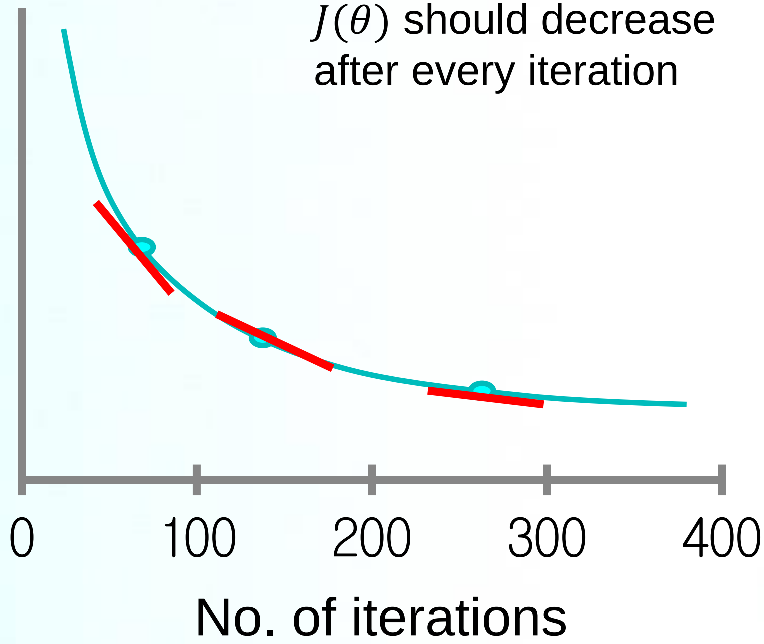
  - "Debugging"
    - How to make sure gradient descent is working correctly

  - How to choose learning rate α

Gradient Descent

- Making sure gradient descent is working correctly

$$\min_{\theta} J(\theta)$$

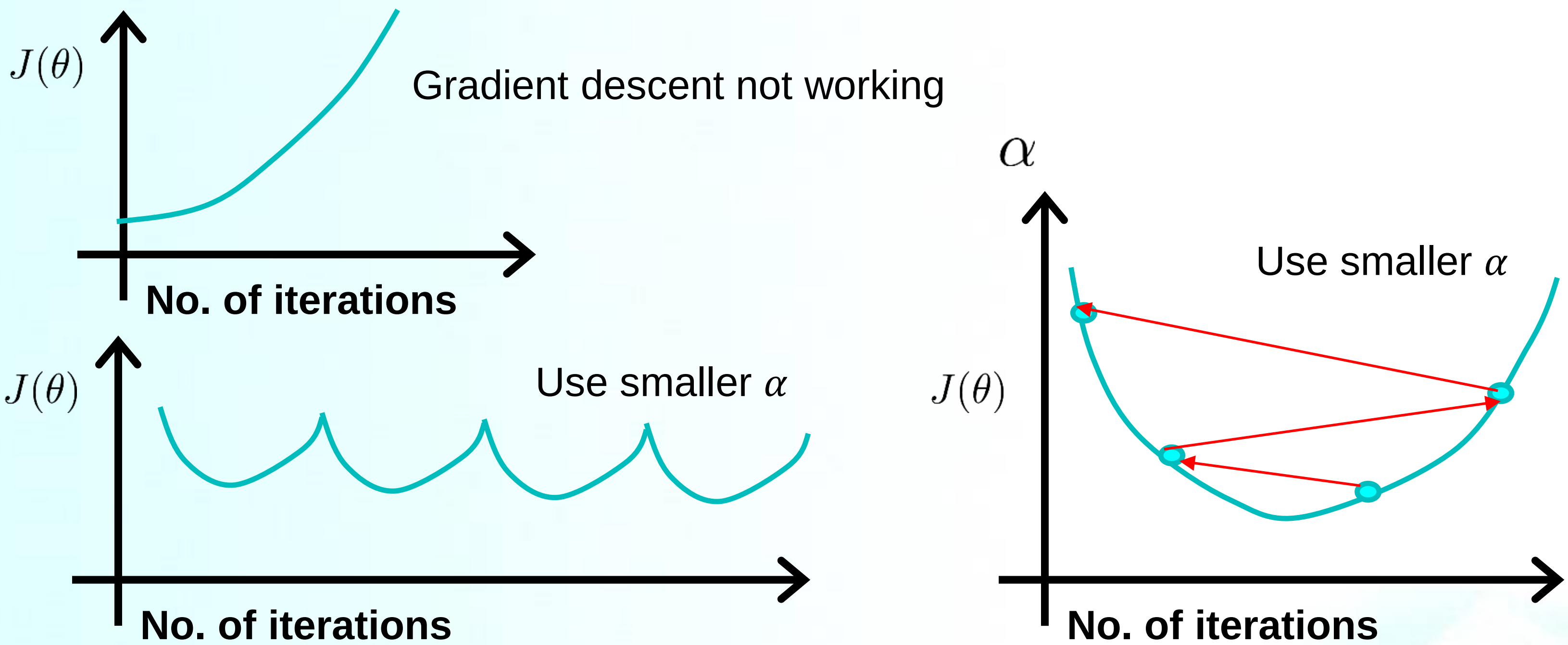


- Declare convergence

  if $J(\theta)$ decreases by less than $\varepsilon = 10^{-3}$ in one iteration.

Gradient Descent

- Making sure gradient descent is working correctly



- For sufficiently small α, $J(\theta)$ should decrease on every iteration.
  - But if α is too small, gradient descent can be slow to converge.

Gradient Descent

- Summary
  - Too small α
    - Slow convergence
  - Too large α
    - Most of times, $J(\theta)$ may not decrease on every iteration
    - $J(\theta)$ may not converge
    - (Sometimes, slow convergence is also possible.)
  - To choose α, try
    - …, 0.001, 0.003, 0.01, 0.03, 0.1, 0.3, 1, …

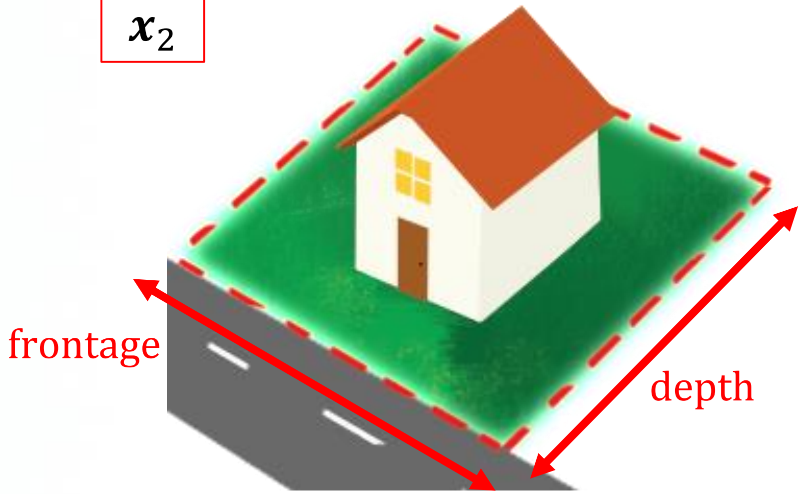
X 3 $\approx$X 3 X 3 $\approx$X 3

Outline

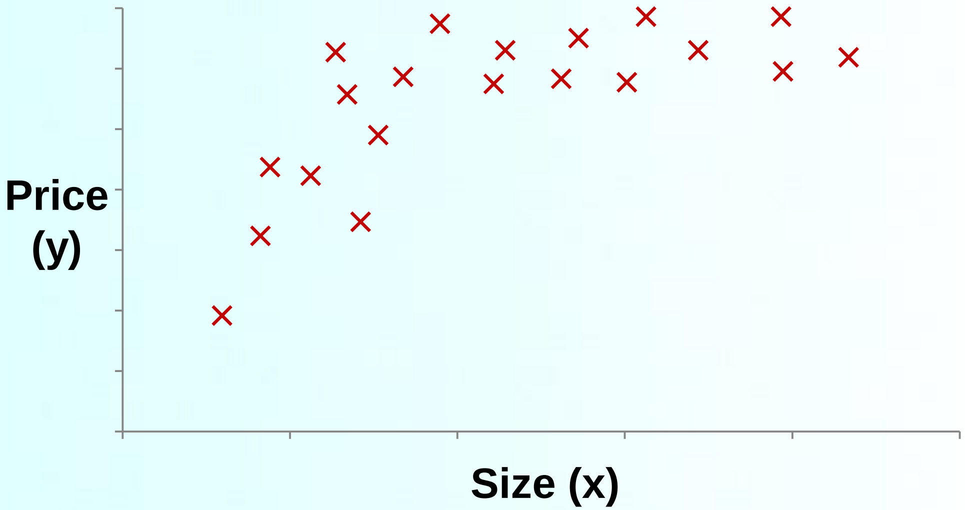
- Multiple features
- Gradient descent for multiple variables
- Feature scaling in gradient descent
- Learning rate in gradient descent
- Features and polynomial regression
- Normal equation

Housing Prices Prediction

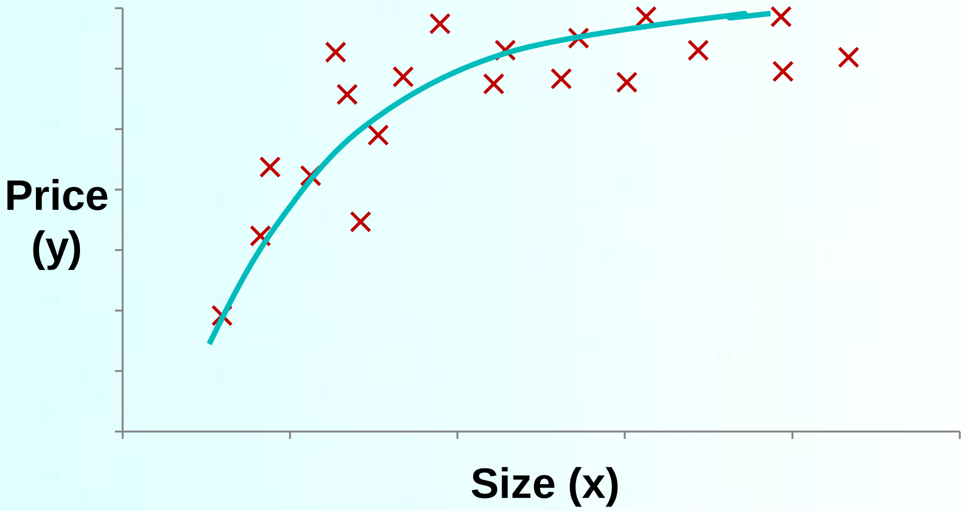
- Two features
  - $h_\theta(x) = \theta_0 + \theta_1 \times \underbrace{frontage}_{\boxed{x_1}} + \theta_2 \times \underbrace{depth}_{\boxed{x_2}}$
- (New) One feature
  - Area x = frontage*depth
  - $h_\theta(x) = \theta_0 + \theta_1 x$



Polynomial Regression

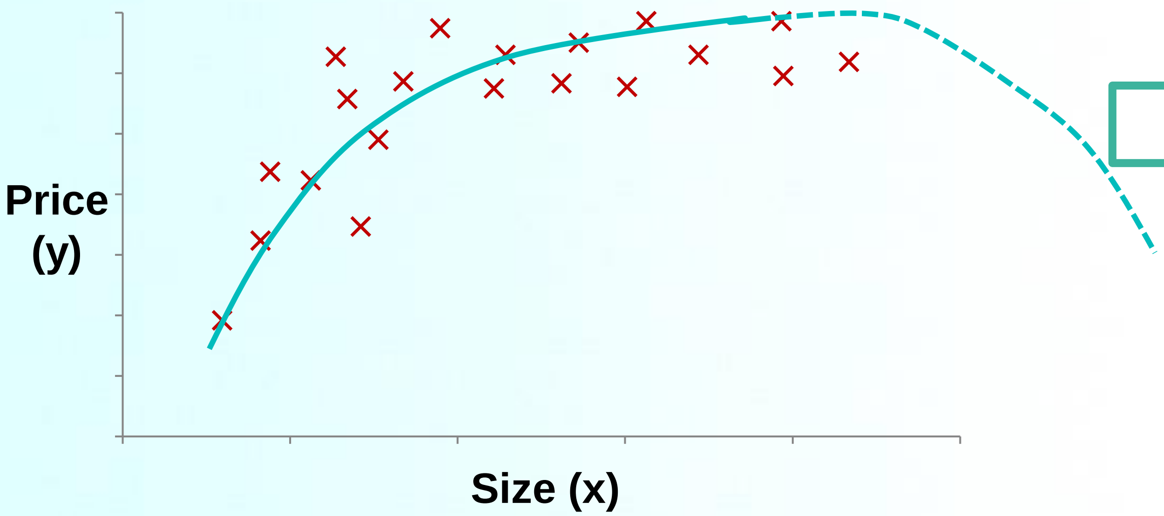


Polynomial Regression



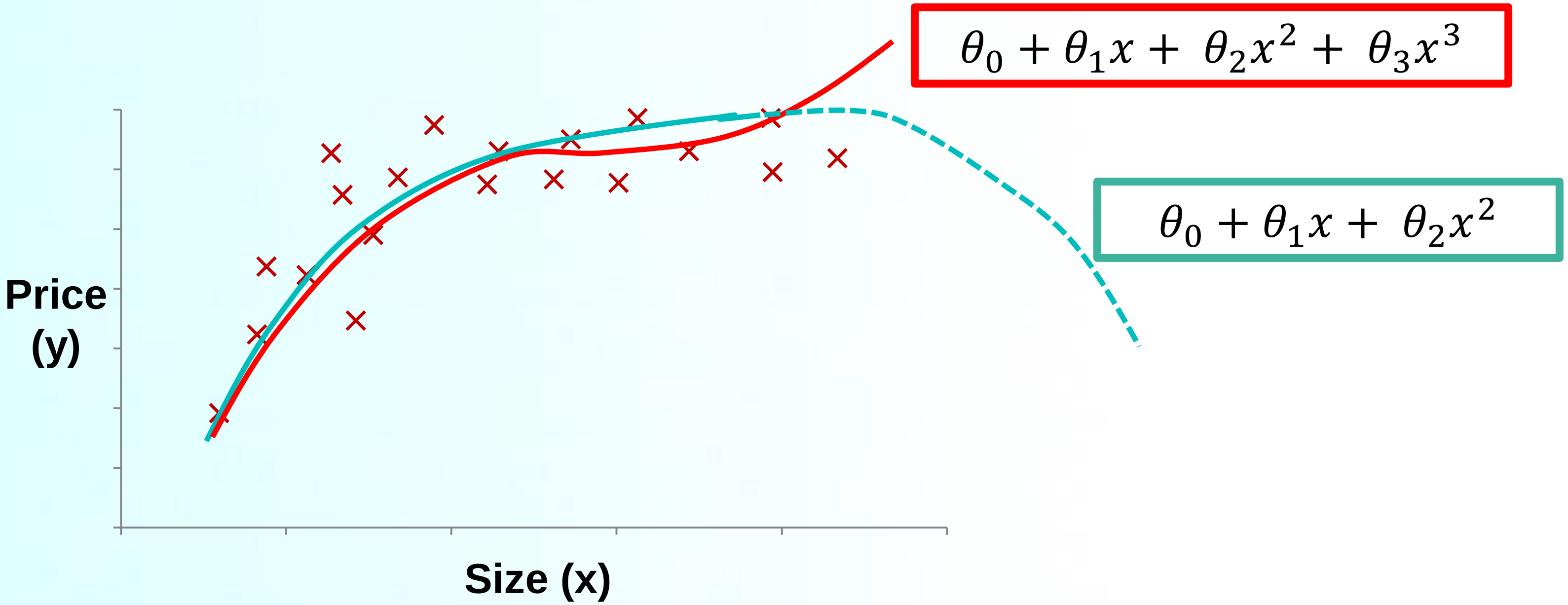
$$\theta_0 + \theta_1 x + \theta_2 x^2$$

Polynomial Regression



$$\theta_0 + \theta_1 x + \ \theta_2 x^2$$

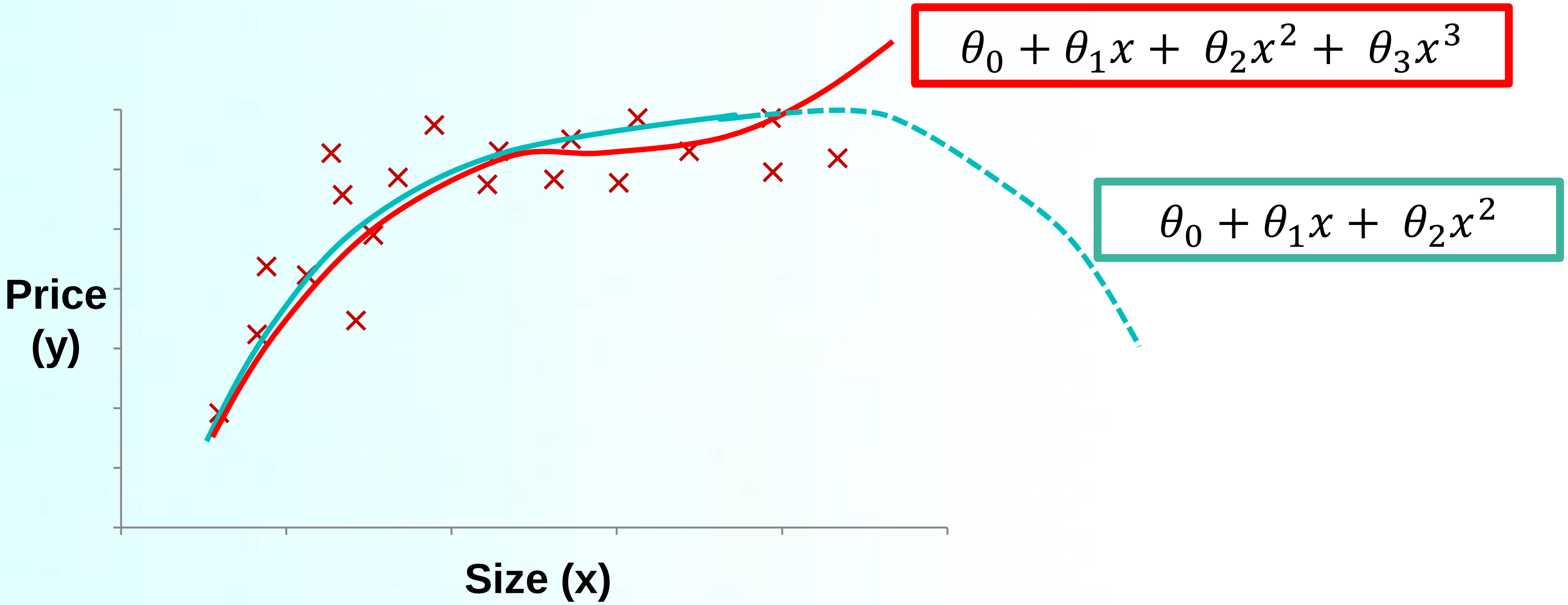
Polynomial Regression



- $h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_3 = \theta_0 + \theta_1(size) + \theta_2(size)^2 + \theta_3(size)^3$

  $x_1 = (size),\ x_2 = (size)^2,\ x_3 = (size)^3$

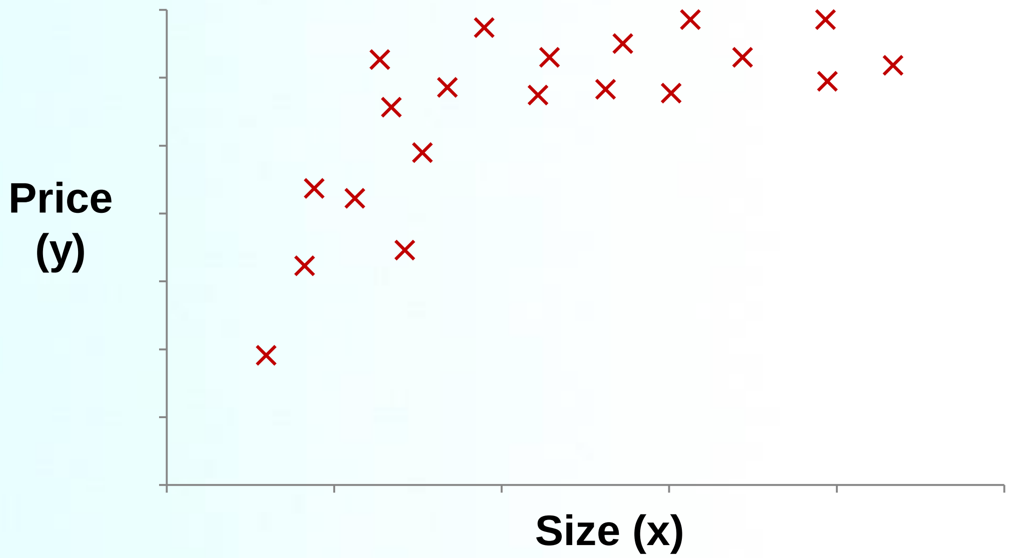
Polynomial Regression



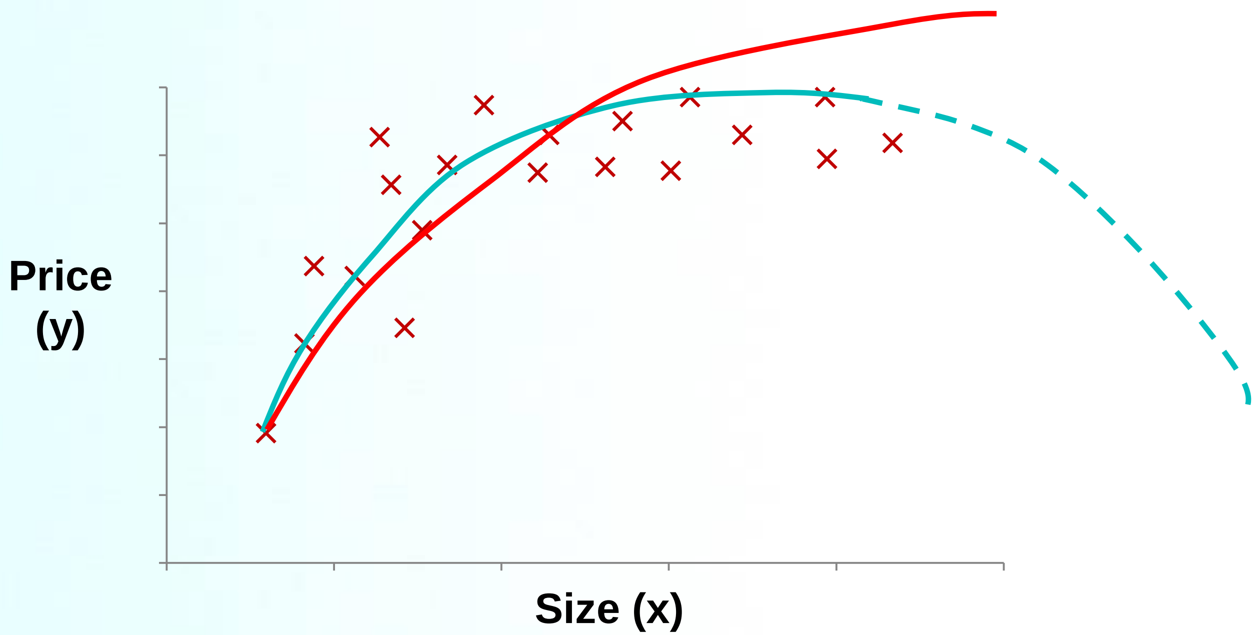
- $h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_3 = \theta_0 + \theta_1 (size) + \theta_2 (size)^2 + \theta_3 (size)^3$
  $x_1 = (size),\ x_2 = (size)^2,\ x_3 = (size)^3$

- Feature scaling is necessary
  - size: 1-1,000 (ft^2)➔ $size^2$: 1~10^6, $size^3$: 1~10^9

Choice of Features



Choice of Features



$$h_\theta(x) = \theta_0 + \theta_1(size) + \theta_2(size)^2$$

$$h_\theta(x) = \theta_0 + \theta_1(size) + \theta_2\sqrt{size}$$

Extending Linear Regression

- Extending Linear Regression to More Complex Models
  - The inputs $\boldsymbol{x}$ for linear regression can be:
    - Original quantitative inputs
    - Transformation of quantitative inputs
      - log, exp, square root, square, etc.
    - Polynomial transformation
      - $y = \beta_0 + \beta_1 \cdot x_1 + \beta_2 \cdot x_2 + \beta_3 \cdot x_3$
    - Basis expansions
    - Dummy coding of categorical inputs
    - Interactions btw variables
      - example: $x_3 = x_1 \cdot x_2$

➔ This allows use of linear regression techniques to fit non-linear datasets

Linear Basis Function Models

- Generally,

$$h_\theta(\boldsymbol{x}) = \sum_{j=0}^{n} \theta_j \boxed{\phi_j(\boldsymbol{x})}$$

Basis function

- Typically, $\phi_0(x) = 1$ so that θ_0 acts as a bias

- In the simplest case, we use linear basis functions:

$$\phi_j(\boldsymbol{x}) = x_j$$

Linear Basis Function Models

- Polynomial basis functions

$$\emptyset_j(\boldsymbol{x}) = x^j$$

- Gaussian basis functions

$$\emptyset_j(\boldsymbol{x}) = exp\left\{-\frac{\left(x-\mu_j\right)^2}{2s^2}\right\}$$

- Sigmoidal basis functions

$$\emptyset_j(\boldsymbol{x}) = \sigma\left(\frac{x-\mu_j}{s}\right)$$

where $\sigma(a) = \frac{1}{1+\exp(-a)}$

Outline

- Multiple features
- Gradient descent for multiple variables
- Feature scaling in gradient descent
- Learning rate in gradient descent
- Features and polynomial regression
- Normal equation

$Ax = b$.

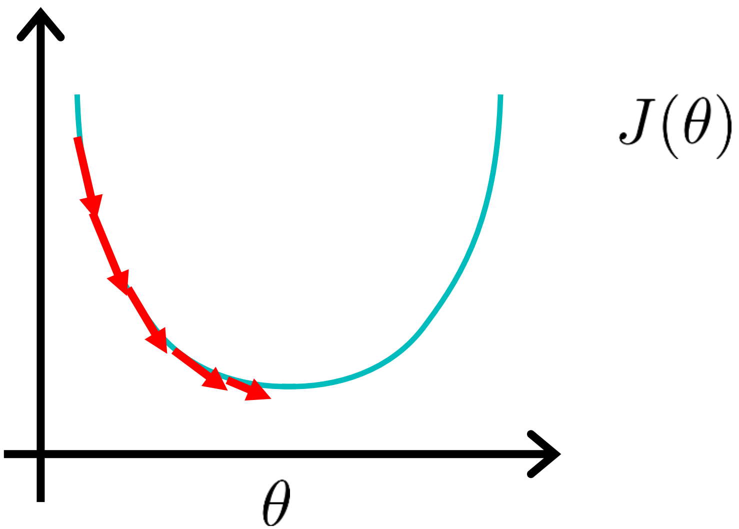
Normal Equation

- A least-square solution $\tilde{v}$ to $Av = w$
  - $\tilde{v} = min_v \|Av - w\|^2$

iff

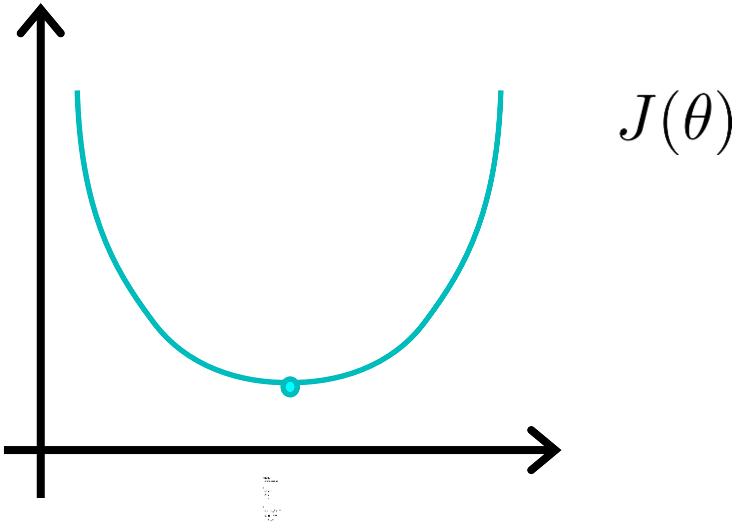
- $\tilde{v}$ is a solution to the normal equation $A^T Av = A^T w$
  - i.e. $\tilde{v} = (A^T A)^{-1} A^T w$

Gradient Descent



Normal Equation

- Normal equation
  - Method to solve for θ analytically



- If $\theta \in R$
  - $J(\theta) = a\theta^2 + b\theta + c$
  - Set $\frac{d}{d\theta}J(\theta) = \ \ldots = 0$
    - Solve for θ

- If $\theta \in \ R^{n+1}$
  - $J(\theta_0, \theta_1, \cdots, \theta_n\ ) = \frac{1}{2m}\sum_{i=1}^{m}\left(h_\theta\left(x^{(i)}\right) - y^{(i)}\right)^2$
  - Set $\frac{\partial}{\partial\theta_j}J(\theta) = \ \ldots = 0$ (for every j)
    - Solve for $\theta_0, \theta_1, \cdots, \theta_n$

Normal Equation

- If $\theta \in R^{n+1}$
  - $J(\theta_0, \theta_1, \cdots, \theta_n) = \frac{1}{2m} \sum_{i=1}^{m} \left(h_\theta\left(x^{(i)}\right) - y^{(i)}\right)^2$
  - $min_\theta J(\theta) = min_\theta \frac{1}{2m} \|X\theta - y\|^2$
    - where $\theta = \begin{bmatrix} \theta_0 \\ \theta_1 \\ \vdots \\ \theta_n \end{bmatrix}$, $X = \begin{bmatrix} \left(x^{(1)}\right)^T \\ \left(x^{(2)}\right)^T \\ \vdots \\ \left(x^{(m)}\right)^T \end{bmatrix}$ (design matrix), $x^{(i)} = \begin{bmatrix} x_0^{(i)} \\ x_1^{(i)} \\ x_2^{(i)} \\ \vdots \\ x_n^{(i)} \end{bmatrix}$

$\rightarrow \theta = (X^T X)^{-1} X^T y$

Example

- $m = 4$

$\left[x^{(2)}\right]^T$

x_0	Size (feet²) x_1	Number of Bedrooms x_2	Number of Floors x_3	Age of home (years) x_4	Price ($1000) y
1	2104	5	1	45	460
1	1416	3	2	40	232
1	1534	3	2	30	315
1	852	2	1	36	178

- $X = \begin{bmatrix} 1 & 2104 & 5 & 1 & 45 \\ 1 & 1416 & 3 & 2 & 40 \\ 1 & 1534 & 3 & 2 & 30 \\ 1 & 852 & 2 & 1 & 36 \end{bmatrix} \in R^{4\times(n+1)}$ $\quad y = \begin{bmatrix} 460 \\ 232 \\ 315 \\ 178 \end{bmatrix}$

- $\theta = (X^T X)^{-1} X^T y$

Example

- $m = 5$

x_0	Size (feet²) x_1	Number of Bedrooms x_2	Number of Floors x_3	Age of home (years) x_4	Price ($1000) y
1	2104	5	1	45	460
1	1416	3	2	40	232
1	1534	3	2	30	315
1	852	2	1	36	178
1	3000	4	1	38	540

- $X = \begin{bmatrix} 1 & 2104 & 5 & 1 & 45 \\ 1 & 1416 & 3 & 2 & 40 \\ 1 & 1534 & 3 & 2 & 30 \\ 1 & 852 & 2 & 1 & 36 \\ 1 & 3000 & 4 & 1 & 38 \end{bmatrix} \in R^{5\times(n+1)}$ $\quad y = \begin{bmatrix} 460 \\ 232 \\ 315 \\ 178 \\ 540 \end{bmatrix}$

- $\theta = (X^T X)^{-1} X^T y$

Examples and Features

- m examples: $\left(x^{(1)},\ y^{(1)}\right),\ \left(x^{(2)},\ y^{(2)}\right),\ \dots,\left(x^{(m)},\ y^{(m)}\right)$
- n features
  - $x^{(i)} = \begin{bmatrix} x_0^{(i)} \\ x_1^{(i)} \\ x_2^{(i)} \\ \vdots \\ x_n^{(i)} \end{bmatrix} \in R^{n+1}$, $X = \begin{bmatrix} \left(x^{(1)}\right)^T \\ \left(x^{(2)}\right)^T \\ \vdots \\ \left(x^{(m)}\right)^T \end{bmatrix} \in R^{m\times(n+1)}$
  - $y = \begin{bmatrix} y^{(1)} \\ y^{(2)} \\ \vdots \\ y^{(m)} \end{bmatrix} \in R^m$ ➔ $\theta = (X^TX)^{-1}X^Ty$

Examples and Features

- m examples: $\left(x^{(1)},\ y^{(1)}\right),\ \left(x^{(2)},\ y^{(2)}\right),\ \ldots,\left(x^{(m)},\ y^{(m)}\right)$

- One feature
  - If $x^{(i)} = \begin{bmatrix} 1 \\ x_1^{(i)} \end{bmatrix} \in R^2,$

$$X = \begin{bmatrix} 1 & x_1^{(1)} \\ 1 & x_1^{(2)} \\ \vdots & \vdots \\ 1 & x_1^{(m)} \end{bmatrix} \in R^{m\times 2}$$

Gradient Descent vs Normal Equation

- m examples and n features

Gradient Descent	Normal Equation
Need to choose α	No need to choose α
Needs many iterations	No need to iterate
	Need to compute $(X^T X)^{-1}$ (➔ $O(n^3)$)
Works well even when n is large	Slow if n is very large

Normal Equation

- $\theta = (X^T X)^{-1} X^T y$
  - What if $X^T X$ is non-invertible? (i.e. $(X^T X)^{-1}$ does not exist)
    - Singular or degenerate
      - Pseudo inverse

- Singular $X^T X$
  - Redundant features (linearly dependent)
    - e.g. x_1 = size in feet^2
    - x_2 = size in m^2
    - $x_1 = (3.28)^2 * x_2$
  - Too many features (e.g. $m \leq n$)
    - Delete some features, or use regularization

References

- Andrew Ng, https://www.coursera.org/learn/machine-learning
- Eric Eaton, https://www.seas.upenn.edu/~cis519
- http://www.holehouse.org/mlclass/04_Logistic_Regression.html